HEWLETT-PACKARD COMPANY
Intellectual Property Administration
P.O. Box 272400
Fort Collins, Colorado 80527-2400

OIPE
OCT 03 2005
IAP25
PATENT & TRADEMARK OFFICE

PATENT APPLICATION

ATTORNEY DOCKET NO. 200301894-1

IFW

IN THE
UNITED STATES PATENT AND TRADEMARK OFFICE

Inventor(s): Samuel H. Duncan et al.

Confirmation No.: 3331

Application No.: 09/944,515

Examiner: Mason, Donna K.

Filing Date: August 31, 2001

Group Art Unit: 2111

Title: PASSIVE RELEASE AVOIDANCE TECHNIQUE

Mail Stop Appeal Brief-Patents
Commissioner For Patents
PO Box 1450
Alexandria, VA 22313-1450

TRANSMITTAL OF APPEAL BRIEF

Sir:

Transmitted herewith is the Appeal Brief in this application with respect to the Notice of Appeal filed on August 2, 2005.

The fee for filing this Appeal Brief is (37 CFR 1.17(c)) $500.00.

(complete (a) or (b) as applicable)

The proceedings herein are for a patent application and the provisions of 37 CFR 1.136(a) apply.

( ) (a) Applicant petitions for an extension of time under 37 CFR 1.136 (fees: 37 CFR 1.17(a)-(d) for the total number of months checked below:

( )	one month	$120.00
( )	two months	$450.00
( )	three months	$1020.00
( )	four months	$1590.00

( ) The extension fee has already been filled in this application.

(X) (b) Applicant believes that no extension of time is required. However, this conditional petition is being made to provide for the possibility that applicant has inadvertently overlooked the need for a petition and fee for extension of time.

Please charge to Deposit Account **08-2025** the sum of **$500.00**. At any time during the pendency of this application, please charge any fees required or credit any over payment to Deposit Account 08-2025 pursuant to 37 CFR 1.25. Additionally please charge any fees to Deposit Account 08-2025 under 37 CFR 1.16 through 1.21 inclusive, and any other sections in Title 37 of the Code of Federal Regulations that may regulate fees. A duplicate copy of this sheet is enclosed.

(X) I hereby certify that this correspondence is being deposited with the United States Postal Service as first class mail in an envelope addressed to: Commissioner for Patents, Alexandria, VA 22313-1450. Date of Deposit: 09/30/2005

OR

( ) I hereby certify that this paper is being transmitted to the Patent and Trademark Office facsimile number________ on ________

Number of pages: 20

Typed Name: Melissa Altman

Signature: Melissa Altman

Respectfully submitted,

Samuel H. Duncan et al.

By Michael R. Reinemann

Michael R. Reinemann
Attorney/Agent for Applicant(s)
Reg. No. **38,280**

Date: **09/30/2005**

Telephone No.: **(617)951-2500**

OIPE
IAP25
OCT 0 3 2005
PATENT & TRADEMARK OFFICE

PATENTS
15311-2292
PD00-3270

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In Re The Application of: Samuel H. Duncan et al.	)	
Serial No.: 09/944,515	)	Examiner: Mason, Donna K.
Filed: August 31, 2001	)	Art Unit: 2111
For: PASSIVE RELEASE AVOIDANCE TECHNIQUE	)	

Cesari and McKenna, LLP
88 Black Falcon Avenue
Boston, MA 02210
September 30, 2005

CERTIFICATE OF MAILING

I hereby certify that the following paper is being deposited with the United States Postal Service as first-class mail in an envelope addressed to the Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450, on September 30, 2005.

Melissa Altman
Melissa L. Altman

_X_ Appeal Brief
_X_ Return Receipt Postcard
_X_ Transmittal of Appeal Brief

PATENTS
15311-2292
200301894-1

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In Re The Application of: Samuel H. Duncan et al.	)	
Serial No.: 09/944,515	)	Examiner: Mason, Donna K.
Filed: August 31, 2001	)	Art Unit: 2111
For: PASSIVE RELEASE AVOIDANCE TECHNIQUE	)	

Cesari and McKenna, LLP
88 Black Falcon Avenue
Boston, MA 02210
September 30, 2005

CERTIFICATE OF MAILING

I hereby certify that the following paper is being deposited with the United States Postal Service as first-class mail in an envelope addressed to the Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450, on September 30, 2005.

Melissa Altman
Melissa Altman

Mail Stop Appeal Brief - Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

APPEAL BRIEF

In response to the Notice of Appeal mailed August 2, 2005, Applicants hereby submit this Appeal Brief.

10/04/2005 HVUONG1 00000034 082025 09944515
01 FC:1402 500.00 DA

REAL PARTY IN INTEREST

The real party in interest is Hewlett-Packard Development Company, L.P. of Houston, Texas.

RELATED APPEALS AND INTERFERENCES

Applicant and their legal representatives know of no related appeals or interferences that will directly affect or be directly affected by or have a bearing on the Board's decision in the present appeal.

STATUS OF CLAIMS

Claims 1-9, and 11-18 are pending in the case. Claims 1-9, and 11-18 stand finally rejected under 35 U.S.C. §103. Due to an inadvertent numbering error, claim 10 was never presented.

A copy of claims 1-9, and 11-18, in their current form, is attached hereto as an Appendix.

STATUS OF AMENDMENTS

No amendments have been filed since the mailing of the Final Rejection on May 3, 2005.

SUMMARY OF CLAIMED SUBJECT MATTER

The summary is set forth in two exemplary embodiments that correspond to independent claims 1 and 12. Discussions about elements and recitations of these claims can be found at least at the cited locations in the specification and drawings.

Independent claim 1 is directed to a method for preventing the passive release of interrupts within a computer system. According to the method, an interrupt signal is asserted

by an input/output (I/O) device coupled to an I/O bridge (block 604). An interrupt message corresponding to that signal is forwarded to a processor for servicing (block 608), and an interrupt pending flag is set (block 612). A first ordered message notifying the I/O device that the interrupt has been serviced is generated and sent (block 616). A second ordered message for clearing the interrupt pending flag is sent after the first message (block 620). The interrupt signal is de-asserted in response to the first message (block 628), and the interrupt pending flag is cleared in response to the second message (block 632). See Specification at pp. 9-13 and Figs. 6A and 6B.

Independent claim 12 is directed to a computer system (100) including a plurality of I/O devices (322, 324) that assert and de-assert interrupt signals, a processor (104) for servicing interrupts, and an I/O bridge (106) that interfaces between the I/O devices and the processor. The I/O bridge includes an interrupt controller (506) that detects the assertion and de-assertion of interrupt signals. In response to detecting an interrupt signal, the interrupt controller is configured to issue an interrupt message to the processor (block 608), and set an interrupt pending flag (530). The processor, upon servicing the interrupt, is configured to send first and second ordered messages to the I/O bridge (blocks 616 and 620). The first ordered message notifies the I/O device that the interrupt has been serviced, while the second ordered message calls for the interrupt pending flag to be cleared. The I/O device is further configured to de-assert the interrupt signal in response to the first ordered message, and the interrupt controlled is configured to clear the interrupt pending flag in response to the second ordered message. See Specification at pp. 4-6 and 8-13, and Figs. 1-3, 5 and 6.

Claim 2 depends from claim 1, and further recites that the first ordered message is forwarded from an I/O bridge port to the subject I/O device (block 626). See Specification at p. 13, and Fig. 6B.

Claim 15 depends from claim 12, and further recites that the I/O bridge port has a read cache (328) to buffer messages from the processor (104), and an ordering engine (408) that is coupled to the read cache. The ordering engine, moreover, is configured to release messages from the read cache in the same order in which they were received. See Specification at pp. 7-8, and Fig. 4.

GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL

Whether claims 1 and 12, which otherwise meet all conditions of patentability under Title 35 of the United States Code, are unpatentable under 35 U.S.C. §103 over U.S. Pat. No. 5,956,516 to Pawlowski ("Pawlowski") in view of the PCI Local Bus Specification, Revision 2.1, June 1, 1995 ("PCI Spec.") or, in the alternative, based on U.S. Pat. No. 6,629,179 to Bashford ("Bashford") in view of the PCI Spec., where the art of record either alone or in combination fails to teach or suggest, among other things, generating a first ordered message after servicing an interrupt indicating that the interrupt has been serviced, or a second ordered message, which is sent after the first ordered message for clearing an interrupt pending flag.

Whether claim 2, which otherwise meets all conditions of patentability under Title 35 of the United States Code, is unpatentable under 35 U.S.C. §103 over Pawlowski in view of the PCI Spec., where the art of record either alone or in combination fails to teach or suggest that the first ordered message is forwarded from an I/O bridge port to the subject I/O device.

Whether claim 15, which otherwise meets all conditions of patentability under Title 35 of the United States Code, is unpatentable under 35 U.S.C. §103 over Pawlowski in view of the PCI Spec., where the art of record either alone or in combination fails to teach or suggest a read cache that buffers messages from the processor, or an ordering engine that releases messages from the read cache in the same order in which they were received.

ARGUMENT

Legal Standard

In rejecting claims under 35 U.S.C. §103, the examiner bears the initial burden of presenting a prima facie case of obviousness. See, e.g., In re Rijckaert, 9 F.3d 1531, 1532 (Fed. Cir. 1993). To establish a prima facie case of obviousness, the references, when considered in their entirety, must teach or suggest all of the claimed limitations. If the references fail to teach or suggest any one of the claimed limitations, then the rejection should be reversed. An examiner may not, moreover, resort to speculation, unfounded assumption or hindsight reconstruction to supply deficiencies in the references. In re Warner, 379 F.2d 1011, 1017 (CCPA 1967). In addition, the Examiner must set forth specific reasons why one skilled in the art would be motivated to select the cited features for combination in the manner claimed. In re Rouffet, 149 F.3d 1350, 1357 (Fed. Cir. 1998). Such a requirement is intended to avoid impermissible hindsight, in which the inventor's own disclosure is used as a blueprint for piecing together the prior art to defeat patentability. In re Dembiczak, 175 F.3d 994, 999 (Fed. Cir. 1999).

The claims do not stand or fall together. Instead Applicants present separate arguments for various independent and dependent claims. Each of these arguments is separately

argued below and presented with separate headings and sub-headings as required by 37 C.F.F. §41.37(c)(1)(vii).

Description of the Cited References

Pawlowski describes an interrupt handling mechanism that allows more peripheral devices to be added to a computer system without having to increase the number of interrupt pins on the interrupt controller. Specifically, Pawlowski discloses a "redirection table" 128, which is located on the interrupt controller 34, and includes a series of "send pending" (SP) bits, which Pawlowski also describes as delivery status bits. The redirection table further includes a series of interrupt request register (IRR) bits.

Bashford describes a system that ensures that any posted write transactions that were pending when an interrupt message is received are written before generating and posting a message signaled interrupt associated with the received interrupt message.

The PCI Spec. is a published specification regarding the Peripheral Component Interface Local Bus. The cited excerpt from the PCI Spec., i.e., Appendix E - System Transaction Order, specifies certain ordering rules to be implemented by PCI compatible bridges to prevent deadlock.

Claims 1 and 12

Claim 1 recites, in relevant part, as follows:

> "A method for preventing passive release of interrupts within a computer system, the computer system having at least one processor for servicing the interrupts, one or more input/output (I/O) devices configured to issue interrupts, and an I/O bridge having a plurality of ports to which I/O devices are coupled and configured to interface between the I/O devices and the processor, the method comprising the steps of:"

"asserting an interrupt signal by a subject I/O device coupled to a given port of the I/O bridge",

"forwarding an interrupt message corresponding to the interrupt signal to the processor for servicing",

"setting an interrupt pending flag in response to assertion of the interrupt signal",

"in response to the interrupt being serviced, generating a first ordered message, the first ordered message notifying the subject I/O device that the interrupt has been serviced",

"generating a second ordered message for clearing the interrupt pending flag",

"sending the first ordered message to the given port of the I/O bridge", and

"sending the second ordered message to the given port of the I/O bridge after the first message has been sent".

As shown, claim 1 calls for a first ordered message to be generated and sent that notifies the I/O device that its interrupt has been serviced. Following the first ordered message, a second ordered message is generated and sent for clearing the interrupt pending flag.

Rejections Based on Pawlowski

The final Office Action contends that Pawlowski's "IRR bit" as described at Col. 6, lines 40-41 corresponds to Applicants' claimed first message, and that the PCI Spec. at p. 258 teaches "ordered messages", thereby rendering claim 1 obvious. Applicants respectfully disagree.

First, Pawlowski's IRR bit is not a message. It is an Interrupt Request Register that is part of a data table, namely a redirection table (128), which is found on Pawlowski's interrupt controller (34). The IRR, moreover, has a plurality of bits. <u>See</u> Pawlowski at Col. 6, lines 37-39, and Figs. 3 and 4 ("redirection table 128 also includes interrupt request register

(IRR) bits, 0, 1, . . . , X-1, which are used in the case of level triggered interrupts"). The excerpt from Pawlowski cited in the final Office Action as purportedly teaching Applicants first ordered message simply describes the circumstances under which a given bit of the IRR is set. In particular, the cited excerpt states:

"The IRR bit is set when an interrupt message is accepted by the processor."

In sum, Pawlowski's IRR is a stationary register used by his interrupt controller to indicate when a level triggered interrupt has been "accepted by the processor."

In contrast, claim 1 recites, among other things, a first ordered **message** that is generated in response to the interrupt **being serviced**, and that is sent to the given port of the I/O bridge to which the subject I/O device is coupled. Nowhere does Pawlowski teach or suggest that his IRR is sent to or from any device, and no such teaching or suggestion is identified by the final Office Action. Instead, Pawlowski describes his IRR as a motionless data store fixedly located at the interrupt controller. Applicants submit that it is improper to characterize Pawlowski's stationary IRR as a message that is generated and sent within a computer system.

In addition, unlike Applicants' first message which notifies the subject I/O device that the interrupt has been serviced, Pawlowski's IRR bit is set when an interrupt message is accepted by the processor. Thus, Pawlowski's IRR bit is presumably set sometime **before** the interrupt is actually serviced.

For these reasons, Applicants submit that Pawlowski fails to teach or suggest a first message generated in response to the interrupt being serviced that notifies the I/O device that

the interrupt has been serviced, and therefore the final Office Action fails to establish a prima facie case of obviousness based on Pawlowski.

Claim 12 similarly recites, among other things:

> "the at least one processor, upon servicing the interrupt, sends first and second ordered messages to the given port of the I/O bridge, **the first ordered message notifying the subject I/O device that the interrupt has been serviced**, and the second ordered message clearing the interrupt pending flag".

For the reasons set forth above, Applicants submit that Pawlowski fails to teach or suggest a message that notifies a subject I/O device that an interrupt has been serviced. Accordingly, the final Office Action similarly fails to establish a prima facie case of obviousness with regard to claim 12.

Rejections Based on Bashford

The final Office Action alternatively contends that Bashford's "sent signal" corresponds to Applicants' claimed first message, and that the PCI Spec. at p. 258 teaches "ordered messages." Once again, Applicants respectfully disagree.

Bashford's "sent signal" is not a message which is generated in response to an interrupt being serviced, and which notifies a subject I/O device that the interrupt has been serviced, as recited in claim 1. Instead, Bashford's "sent signal" simply indicates that a message signaled interrupt has been sent. Bashford provides no teaching or suggestion that his "sent signal" somehow notifies an I/O device that its interrupt has been serviced.

In particular, at Col. 9, lines 50-64, which is the excerpt relied upon in the final Office Action, Bashford states:

> Then in operation 1010, it is determined whether the message signaled interrupt has been sent out to the primary PCI interface 110. For example, **a message sig-**

> **naled interrupt is sent if the primary PCI interface 110 returns a "sent" signal to the interrupt message generator 410**. In such cases, the request to the PCI interface 110 is deasserted and the interrupt pending flag for the transmitted interrupt bit number is cleared in operation 1012. However, if the message signaled interrupt has not been sent out, the interrupt message generator 410 waits until it has been sent out. After clearing the interrupt pending flag, the current interrupt pointer 834 of the circular queue 408 is incremented to point to the next register location for generating the next message signaled interrupt for transmission. The method then terminates in operation 1016.

As shown, what Bashford teaches is that a primary PCI interface (which interfaces to a host processor) returns a "sent signal" when it has sent a message signaled interrupt to the host processor. The issuance of Bashford's sent signal has no relation to an interrupt being serviced. Instead, it simply confirms that a message signaled interrupt has been sent, presumably to be serviced some time in the future.

In contrast, claim 1 explicitly recites that a first ordered message is generated in response to an interrupt being serviced, and the first ordered message notifies the subject I/O device that the interrupt has been serviced. There is no teaching or suggestion by Bashford that his sent signal somehow notifies an I/O device that its interrupt has been serviced. For example, Bashford's sent signal is never conveyed to an I/O device. Rather, it is a signal that remains internal to the PCI bridge. Bashford's sent signal moreover, is merely used to confirm that a message signaled interrupt has indeed been received by the primary PCI interface, and sent to the host processor for servicing. Accordingly, Applicants submit that the final Office Action fails to establish a prima facie case of obviousness based on Bashford.

Furthermore, far from notifying an I/O device that its interrupt has been serviced, as is the case with Applicants' "first ordered message," the issuance of Bashford's "sent signal" results in an interrupt pending flag being cleared. With Applicant's invention, it is a second

ordered message, issued <u>after</u> the first message, that causes an interrupt pending flag to be cleared. Accordingly, by disclosing a system in which an interrupt pending flag is cleared once a message signaled interrupt has been sent out for processing, Bashford actually teaches away from the present invention, which calls for the second of two ordered messages to clear an interrupt pending flag.

Because Bashford fails to teach or suggest a first message generated in response to an interrupt being serviced that notifies an I/O device that the interrupt has been serviced, as recited in claims 1 and 12, the final Office Action fails to establish a <u>prima facie</u> case of obviousness based on Bashford.

Lack of Motivation to Combine Cited References

According to the final Office Action, the motivation for combining Pawlowski and the PCI Spec. is "to prevent deadlock of the system bus". <u>See</u> final Office Action, p. 7. However, Pawlowski never suggests that deadlock is a problem with either the prior art systems, or with his own system. Instead, the problem identified by Pawlowski is the handling of interrupts from an increasing number of peripheral devices, given the limited number of interrupt pins on an interrupt controller. <u>See</u> Col. 1, line 66 to Col. 2, line 6. Pawlowski nowhere mentions any issue regarding deadlock of the system bus. Thus, if one skilled in the art were seeking to solve the problem of deadlock, there is no reason why he or she would be motivated to consider Pawlowski, which has nothing to do with deadlock avoidance.

This same motivation is also cited as the motivation for combing Bashford with the PCI Spec. <u>See</u> final Office Action at p. 9. Bashford, however, is concerned with errors that might occur when a message signaled interrupt is written before all posted writes that were

pending when the interrupt occurred have completed. See Col. 2, lines 11-14; and Col. 9, lines 47-50. Bashford discloses a system that ensures such writes complete before the interrupt is generated and posted.

Accordingly, Applicants submit that the final Office Action fails to satisfy the requirement of establishing a motivation to combine the cited references in the manner claimed.

For the reasons set forth above, the rejections of claims 1 and 12 should be reversed.

Claim 2

Claim 2, which depends from claim 1, recites "forwarding the first ordered message from the given I/O bridge port to the subject I/O device". As set forth above, the final Office Action equates Pawlowski's IRR with the claimed first ordered message, and further asserts that Pawlowski's IRR teaches the limitation of claim 2. See final Office Action at p. 4. Applicants respectfully disagree.

As noted above, Pawlowski's IRR, which constitutes part of Pawlowski's redirection table 128, is a stationary data structure that remains at all times at the host bridge 16. See Pawlowski at Col. 4, lines 25-27 and Figs. 2 and 4 (which show the redirection table 128 disposed at the host bridge 16). The redirection table 128 is never conveyed or transmitted by the host bride 16 to any other entity in Pawlowski's system, and Pawlowski provides no teaching or suggestion for forwarding either the IRR or the redirection table to any I/O device. Accordingly, the rejection of claim 2 based on Pawlowski should be reversed.

Claim 15

Claim 15, which depends from claim 12, recites:

"the I/O bridge port includes a read cache for buffering messages received from the at least one processor, and an ordering engine operatively coupled to a read cache," and

"the ordering engine is configured to release ordered messages buffered in the read cache in the same order as which they were received".

The final Office Action relies on Pawlowski at Col. 4, lines 56-60 as purportedly teaching or suggesting these limitations. Applicants' respectfully disagree.

This excerpt from Pawlowski states as follows:

> Host bridge 16 may include queues (not shown) to hold various interrupt signals and other signals. Interrupt controller 34 may include queues to hold interrupt request signals. Control logic 130 assists in various functions of interrupt controller 34.

Applicants agree that Pawlowski teaches the use of queues at its host bridge. However, Pawlowski fails to teach or suggest an ordering engine coupled to a read cache that is configured to release messages "buffered in the read cache **in the same order as which they were received**". (emphasis added) The cited excerpt of Pawlowski provides no teaching or suggestion regarding an ordering engine that releases messages in the same order in which they were received. Given Pawlowski's failure to mention any ordering mechanism, only impermissible hindsight could support the contention that this excerpt nonetheless teaches Applicants' claimed read cache and ordering engine. Because Pawlowski fails to teach or suggest a read cache and ordering engine as claimed, the rejection of claim 15 should be reversed.

CONCLUSION

Applicants respectfully submit that the claims are allowable over the art of record. Accordingly, Applicants request that the rejection of all claims be reversed.

Respectfully submitted,

Michael R. Reinemann
Reg. No. 38,280
Tel. 617-951-3060

Please direct all correspondence to:

IP Administration Legal Department,
M/S 35
Hewlett-Packard Co.
P.O. Box 272400
Fort Collins, CO 80527-2400

CLAIMS APPENDIX
(Claims on Appeal in Appl. Ser. No. 09/428,384)

1. (Previously presented) A method for preventing passive release of interrupts within a computer system, the computer system having at least one processor for servicing the interrupts, one or more input/output (I/O) devices configured to issue interrupts, and an I/O bridge having a plurality of ports to which I/O devices are coupled and configured to interface between the I/O devices and the processor, the method comprising the steps of:

asserting an interrupt signal by a subject I/O device coupled to a given port of the I/O bridge;

forwarding an interrupt message corresponding to the interrupt signal to the processor for servicing;

setting an interrupt pending flag in response to assertion of the interrupt signal;

in response to the interrupt being serviced, generating a first ordered message, the first ordered message notifying the subject I/O device that the interrupt has been serviced;

generating a second ordered message for clearing the interrupt pending flag;

sending the first ordered message to the given port of the I/O bridge;

sending the second ordered message to the given port of the I/O bridge after the first message has been sent;

deasserting the interrupt signal in response to the first message; and

clearing the interrupt pending flag in response to the second ordered message.

2. (Original) The method of claim 1 further comprising the step of forwarding the first ordered message from the given I/O bridge port to the subject I/O device.

3. (Original) The method of claim 2 wherein the step of deasserting the interrupt signal is performed by the subject I/O device following its receipt of the first ordered message.

4. (Previously presented) The method of claim 3 wherein one of the ports of the I/O bridge is an interrupt port and the interrupt pending flag is disposed at the interrupt port, the method further comprising the step of forwarding the second ordered message from the given I/O bridge port to the interrupt port.

5. (Original) The method of claim 4 wherein

the interrupt pending flag is implemented through a register of the interrupt port; and

the second ordered message is a write transaction to the register for clearing the interrupt pending flag.

6. (Original) The method of claim 5 further comprising the steps of:

periodically collecting a set of information regarding the assertion of interrupt signals by I/O devices; and

after the step of clearing the interrupt pending flag, waiting a predetermined time before collecting a next set of information regarding the assertion of interrupt signals.

7. (Original) The method of claim 6 wherein the step of periodically collecting is performed through one or more serial data transfer operations.

8. (Original) The method of claim 1 wherein the steps of generating the first and second ordered messages are performed by the processor.

9. (Original) The method of claim 8 wherein the computer system includes (1) a plurality of processors at least one of which is designated to service interrupts from the subject I/O device, and (2) a plurality of I/O bridges each I/O bridge coupled to a plurality of I/O devices configured to assert respective interrupt signals.

10. (Not entered)

11. (Original) The method of claim 1 wherein the interrupt signals are level sensitive interrupts (LSIs).

12. (Previously presented) A computer system comprising:

a plurality of input/output (I/O) devices configured to assert and deassert respective interrupt signals;

at least one processor for servicing interrupts from the I/O devices; and

an I/O bridge configured to interface between the I/O devices and the at least one processor, the I/O bridge having a plurality of ports to which the I/O devices are coupled and an interrupt controller configured to detect the assertion and deassertion of the interrupt signals, wherein

the interrupt controller, in response to assertion of an interrupt signal by a subject I/O device coupled to a given port of the I/O bridge, issues an interrupt message to the at least one processor and sets an interrupt pending flag;

the at least one processor, upon servicing the interrupt, sends first and second ordered messages to the given port of the I/O bridge, the first ordered message notifying the subject I/O device that the interrupt has been serviced, and the second ordered message clearing the interrupt pending flag;

the subject I/O device deasserts the interrupt signal in response to the first message; and

the interrupt pending flag is cleared in response to the second ordered message.

13. (Previously presented) The computer system of claim 12 wherein

one of the ports of the I/O bridge is an interrupt port at which the interrupt controller is disposed, and

the given port of the I/O bridge forwards the second ordered message to the interrupt port after forwarding the first ordered message to the subject I/O device.

14. (Original) The computer system of claim 13 wherein the interrupt port of the I/O bridge includes at least one register at which the interrupt pending flag is implemented.

15. (Original) The computer system of claim 12 wherein

the I/O bridge port includes a read cache for buffering messages received from the at least one processor, and an ordering engine operatively coupled to a read cache, and

the ordering engine is configured to release ordered messages buffered in the read cache in the same order as which they were received.

16. (Previously presented) The computer system of claim 12 further comprising an interrupt collector having a parallel-load shift register for receiving the interrupt signals from the I/O devices, the parallel-load shift register configured to transfer information indicating the assertion or deassertion of interrupt signals to the interrupt controller through one or more serial shift operations.

17. (Original) The computer system of claim 16 wherein

the interrupt collector transfers the information in response to a request from the interrupt controller, and

the interrupt controller is configured to limit the number of serial shift operations performed by the interrupt collector so as to receive only information associated with interrupt signals that have been enabled.

18. (Original) The computer system of claim 12 wherein the interrupt signals are level sensitive interrupts (LSIs).

EVIDENCE APPENDIX

None.

RELATED PROCEEDINGS APPENDIX

None.